SOLVING LARGE MARGIN ESTIMATION OF HMMS VIA SEMIDEFINITE PROGRAMMING

Xinwei Li, Hui Jiang

Department of Computer Science and Engineering, York University,
4700 Keele Street, Toronto, Ontario M3J 1P3, CANADA
Email: {xwli, hj}@cs.yorku.ca

ABSTRACT

In this paper, we propose to use a new optimization method, i.e., *semidefinite programming (SDP)*, to solve large margin estimation (LME) problem of continuous density hidden Markov models (CDHMM) for speech recognition. First of all, we introduce a new constraint into the LME to guarantee the boundedness of the margin of CDHMM. Secondly, we show that the LME problem under this new constraint can be formulated as an SDP problem under some relaxation conditions and it can be solved very efficiently by using some fast optimization algorithms specially designed for SDP. The new method is evaluated in a continuous digit string recognition task by using the TIDIGITS database. Experimental results clearly demonstrate that the new SDP-based method outperforms the previously proposed optimization methods using gradient descent search in both recognition accuracy and convergence speed. With the SDP-based optimization method, the best LME models achieves **0.53%** in string error rate and **0.18%** in WER on the TIDIGITS task. To our best knowledge, this is the best result ever reported in this task.

Index Terms: large margin estimation, CDHMM, semidefinite programming, speech recognition, discriminative training.

1. INTRODUCTION

Recently, we have proposed the large margin estimation (LME) of HMMs for speech recognition [1, 2, 3, 4], where continuous density hidden Markov models (CDHMM) are estimated based on the large margin principle. As shown in [1, 3], the estimation of large margin CDHMMs turns out to be a minimax optimization problem. However, maximization of margin for CDHMMs may become unbounded unless we impose additional constraints onto the optimization procedure. In [1], a heuristic method, called *iterative localized optimization*, is used to guarantee the existence of an optimal point. In [2], we replace the original definition of margin with relative separation margin which is bounded by definition. In [3], some theoretically-sound constraints are introduced into the minimax optimization to guarantee the boundedness of the margin in LME; Then a gradient descent method called *constrained joint optimization method* is proposed to solve the constrained minimax optimization approximately. Although the constrained minimax problem in *constrained joint optimization method* can be converted into an unconstrained minimization problem as in [3, 4] by casting the constraints as the penalty terms in the objective function, it's still a non-convex nonlinear optimization problem. There is no efficient algorithm available to solve this optimization problem. The gradient descent method used in [3, 4] can only lead to a locally optimal solution which highly depends on the initial models used for the optimization. Moreover, the gradient descent search is hard to control in practice since there are a number of sensitive parameters we need to manually tune for various experimental settings, such as the penalty coefficients and step size and so on.

In this paper, we propose to use a better optimization method for LME of CDHMM in speech recognition. First of all, we introduce a new constraint to bound the margin of CDHMM in LME. Under this new constraint, the LME problem can be easily converted into a *semi-definite programming (SDP)* problem under some relaxation conditions if we adopt the Viterbi approximation in HMM calculation. In this way, we are able to take advantage of the efficient algorithms [5] for SDP to solve the LME of CDHMM for speech recognition. SDP is currently considered as an active area in optimization due to the discovery of new applications in several areas as well as the development of some new efficient algorithms [5]. SDP is an extension of linear programming (LP). It has been shown that most interior-point methods for LP can be generalized to SDP problems. As in LP, these algorithms possess polynomial worst-case complexity under certain computation models. They usually perform very well in practice in terms of efficiency. More importantly, these algorithms can lead to the globally optimal solution since the SDP is a well-defined convex optimization problem. The large margin HMM-based classifiers estimated with the new SDP-based optimization method are evaluated in a continuous digit string recognition task by using the TIDIGITS database. Experimental results show that the newly proposed SDP method is very effective in terms of recognition accuracy and optimization efficiency. The SDP-based optimization yields significantly better performance than the previously proposed gradient descent based methods in [1, 3, 4]. With the SDP-based optimization method, the LME models achieves **0.53%** in string error rate and **0.18%** in WER on the TIDIGITS task. To our best knowledge, this is the best result ever reported in this task.

2. LARGE MARGIN HMM

From [1, 2, 3, 4], we know that the separation margin for a speech utterance X_i in a multi-class classifier can be defined as:

$$\begin{aligned} d(X_i) &= \mathcal{F}(X_i|\lambda_{W_i}) - \max_{j\in\Omega\ j\neq W_i} \mathcal{F}(X_i|\lambda_j) \\ &= \min_{j\in\Omega\ j\neq W_i} [\mathcal{F}(X_i|\lambda_{W_i}) - \mathcal{F}(X_i|\lambda_j)] \end{aligned} \quad (1)$$

where Ω denotes the set of all possible words, λ_W denotes the HMM representing the word W, W_i is the true word identity for X_i and $\mathcal{F}(X|\lambda_W)$ is called discriminant function. Usually, the

discriminant function is calculated in the logarithm scale: $\mathcal{F}(X|\lambda_W) = \log[p(W) \cdot p(X|\lambda_W)]$. In this work, we are only interested in estimating HMMs λ_W and assume $p(W)$ is fixed.

Given a set of training data $\mathcal{D} = \{X_1, X_2, \cdots, X_N\}$, we usually know the true word identities for all utterances in $\mathcal{D}$, denoted as $\mathcal{L} = \{W_1, W_2, \cdots, W_N\}$. The *support vector set* $\mathcal{S}$ is defined as:

$$\mathcal{S} = \{X_i \mid X_i \in \mathcal{D} \text{ and } 0 \leq d(X_i) \leq \gamma\} \tag{2}$$

where $\gamma > 0$ is a pre-set positive number. All utterances in $\mathcal{S}$ are relatively close to the classification boundary even though all of them locate in the right decision regions.

The large margin principle leads to estimating the HMM models Λ based on the criterion of maximizing the minimum margin of all support tokens, which is named as large margin estimation (LME) of HMM.

$$\begin{aligned}\tilde{\Lambda} &= \arg\max_{\Lambda} \min_{X_i \in \mathcal{S}} d(X_i) \\ &= \arg\min_{\Lambda} \max_{X_i \in \mathcal{S}} \max_{j \in \Omega, j \neq W_i} [\mathcal{F}(X_i|\lambda_j) - \mathcal{F}(X_i|\lambda_{W_i})] \end{aligned} \tag{3}$$

Note that the support token set $\mathcal{S}$ is selected and used in LME because most of the other training data with larger margin is usually inactive in optimization towards maximizing the minimum margin.

3. A NEW CONSTRAINT FOR LME OF CDHMMS

As shown in [1, 2, 3], the margin as defined in eq.(1) is actually unbounded for CDHMMs. In other words, we can adjust CDHMM parameters in a way to increase margin unlimitedly. In [3, 4], we introduced some theoretically sound constraints to ensure the existence of the optimal point in the large margin estimation of eq.(3). However, it seems very hard to formulate the constrained minimax optimization in [3, 4] into an SDP problem. Therefore, in this section, we introduce a new constraint under which the large margin HMM problem can be easily converted into an SDP problem.

At first, we assume each speech unit is modeled by an N-state CDHMM with parameter vector $\lambda = (\pi, A, \theta)$, where π is the initial state distribution, $A = \{a_{ij} | 1 \leq i, j \leq N\}$ is transition matrix, and θ is parameter vector composed of mixture parameters $\theta_i = \{\omega_{ik}, \mu_{ik}, \Sigma_{ik}\}_{k=1,2,\cdots,K}$ for each state i, where K denotes number of Gaussian mixtures in each state. The state observation p.d.f. is assumed to be a mixture of multivariate Gaussian distributions with diagonal covariance matrices:

$$\begin{aligned} p(\mathbf{x}|\theta_i) &= \sum_{k=1}^{K} \omega_{ik} \cdot \mathcal{N}(\mathbf{x} \mid \mu_{ik}, \Sigma_{ik}) \\ &= \sum_{k=1}^{K} \omega_{ik} \prod_{d=1}^{D} \sqrt{\frac{1}{2\pi\sigma_{ikd}^2}} e^{-\frac{(x_d - \mu_{ikd})^2}{2\sigma_{ikd}^2}} \end{aligned} \tag{4}$$

where mixture weights ω_{ik}'s satisfy the constraint $\sum_{k=1}^{K} \omega_{ik} = 1$ and $\Sigma_{ik} = \text{diag}(\sigma_{ik1}^2, \sigma_{ik2}^2, \cdots, \sigma_{ikD}^2)$ denotes the diagonal covarance matrix of kth Gaussian in state i. For simplicity, we only consider to estimate mean vectors with the LME method.

Given any speech utterance $X = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_R\}$ and any model λ_k, as shown in [3], under Viterbi approximation, the discriminant function, $\mathcal{F}(X|\lambda_k)$, can be expressed as follows:

$$\mathcal{F}(X|\lambda_k) \approx c - \frac{1}{2} \sum_{t=1}^{R} \sum_{d=1}^{D} \frac{(x_{td} - \mu_{s_t l_t d})^2}{\sigma_{s_t l_t d}^2} \tag{5}$$

where we denote the optimal Viterbi path as $\mathbf{s} = \{s_1, s_2, \cdots, s_R\}$ and $\mathbf{l} = \{l_1, l_2, \cdots, l_R\}$, and $\{\mu_{s_t l_t}, \sigma_{s_t l_t}^2\}$ represent mean and variance vectors of the Gaussian at the time instant t along the optimal path, and c stands for a constant independent from Gaussian mean vectors.

Suppose there are totally L Gaussian mixtures in the CDHMM set. For notational convenience, we denote them as $\mathcal{N}(u_k, \Sigma_k)$ where $k \in (1, 2, \ldots, L)$. Thus we can rewrite eq.(5) as

$$\mathcal{F}(X|\lambda_k) \approx c - \frac{1}{2} \sum_{t=1}^{R} \sum_{d=1}^{D} \frac{(x_{td} - \mu_{k_t d})^2}{\sigma_{k_t d}^2} \tag{6}$$

where we denote the optimal Viterbi path as $\mathbf{k} = \{k_1, k_2, \cdots, k_R\}$, and $\{\mu_{k_t}, \sigma_{k_t}^2\}$ represent mean and variance vectors of the Gaussian at the time instant t along the optimal path.

As a result, the decision margin d_{ij} in eq.(1) can be represented as a standard diagonal quadratic form:

$$\begin{aligned} d_{ij}(X_i) &= \mathcal{F}(X_i|\lambda_{W_i}) - \mathcal{F}(X_i|\lambda_j) \\ &\approx c_{ij} - \frac{1}{2} \sum_{t=1}^{R} \sum_{d=1}^{D} \Big[\frac{(x_{itd} - \mu_{i_t d})^2}{\sigma_{i_t d}^2} - \frac{(x_{itd} - \mu_{j_t d})^2}{\sigma_{j_t d}^2} \Big] \end{aligned} \tag{7}$$

where we denote the optimal Viterbi path against λ_{W_i} as $\mathbf{i} = \{i_1, i_2, \cdots, i_R\}$, and the optimal Viterbi path against λ_j as $\mathbf{j} = \{j_1, j_2, \cdots, j_R\}$, and c_{ij} is a constant.

Obviously, if every term in the summation in eq. (7) is bounded, the margin $d_{ij}(X_i)$ will be bounded. It is easy to see that all these items will be bounded if every mean μ_k in the HMMs is constrained in a limited range. Therefore, we introduce the following spherical constraint for all Gaussian means:

$$R(\Lambda) = \sum_{k=1}^{L} \sum_{d=1}^{D} \frac{(\mu_{kd} - \mu_{kd}^{(0)})^2}{\sigma_{kd}^2} \leq r^2 \tag{8}$$

where r is a pre-set constant, and $\mu_{kd}^{(0)}$ is also a constant which is set to be the value of μ_{kd} in the initial models.

Actually, boundedness of the margin $d(X_i)$ is guaranteed by the following theorem :

Theorem 3.1 *Assume we have a set of CDHMMs, $\Lambda = \{\lambda_1, \lambda_2, \cdots, \lambda_M\}$ and a set of training data, denoted as $\mathcal{D} = \{X_1, X_2, \cdots, X_N\}$. The margin $d(X_i)$, as defined in eq.(1), is bounded for any token X_i in the training set $\mathcal{D}$ as long as the constraint in eq.(8) holds.*

The proof is quite straightforward and can be found in [7].

According to theorem 3.1, the minimum margin in eq.(3) is a bounded function of model parameter set, Λ, under the constraint specified in eq. (8). Therefore, the minimax optimization problem in eq.(3) becomes solvable under the additional constraint eq.(8). If we introduce a new variable $-\rho$ ($\rho > 0$) as the common upper bound for all terms in the minimax optimization, we can convert the minimax optimization in eq.(3) into an equivalent minimization problem as follows:

Problem 1

$$\tilde{\Lambda} = \arg\min_{\Lambda} -\rho \tag{9}$$

subject to

$$\mathcal{F}(X_i|\lambda_j) - \mathcal{F}(X_i|\lambda_{W_i}) \leq -\rho \tag{10}$$

$$R(\Lambda) = \sum_{k=1}^{L}\sum_{d=1}^{D}\frac{(\mu_{kd}-\mu_{kd}^{(0)})^2}{\sigma_{kd}^2} \le r^2 \quad (11)$$

$$\rho \ge 0. \quad (12)$$

for all $X_i \in \mathcal{S}$ and $j \in \Omega$ and $j \neq W_i$. Here r is a pre-set constant. $\mu_{kd}^{(0)}$ is also a constant which is set to the original value of μ_{id} in the initial models.

4. SEMIDEFINITE PROGRAMMING METHOD

In this work, we study how to solve the above LME of CDHMMs with the SDP method. We know that the following form is a standard SDP problem.

$$\text{Minimize} \qquad \sum_{j=1}^{n_b} C_j \cdot X_j \quad (13)$$

subject to

$$\sum_{j=1}^{n_b} A_{i,j}\cdot X_j \le b_i, \qquad i=1,\ldots,m, \qquad X_j \succeq 0. \quad (14)$$

where $X_j \succeq 0$ means each variable X_j is a positive semidefinite matrix. $A_{i,j}$, C_j are real symmetric matrices with the same dimension as X_j, b_i is a scalar constant, and $X \cdot Y$ denotes the inner product of two symmetric matrices as: $X \cdot Y = \text{tr}(X^T Y) = \sum_{i,j} x_{ij}y_{ij}$.

First of all, we introduce some notations: e_i is an L-dimensional vector with -1 at the i-th position, and zero everywhere else. A column vector x is written as $x = (x_1; x_2; \ldots; x_n)$ and a row vector as $x = (x_1, x_2, \ldots, x_n)$. I_D is a $D \times D$ identity matrix. And U is a matrix by concatenating all normalized Gaussian mean vectors as its columns as:

$$U = (\tilde{\mu}_1, \tilde{\mu}_2, \ldots, \tilde{\mu}_L) \quad (15)$$

where each column

$$\tilde{\mu}_k \;:=\; (\mu_{k1}/\sigma_{k1};\ \mu_{k2}/\sigma_{k2};\ \ldots;\ \mu_{kD}/\sigma_{kD}). \quad (16)$$

In the following, we will consider how to convert the minimization **Problem 1** into an SDP as shown in eq.(13).

Firstly, we will re-formulate the constraint in eq.(10) into the standard constraint form eq.(14) in SDP . After some math manipulations, we can re-write eq.(6) as:

$$\begin{aligned}\mathcal{F}(X|\lambda_W) &= c - \frac{1}{2}\sum_{t=1}^{T}(\tilde{x}_t - \tilde{\mu}_{k_t})^T(\tilde{x}_t - \tilde{\mu}_{k_t}) \\ &= c - \frac{1}{2}\sum_{t=1}^{T}(\tilde{x}_t; e_{k_t})^T (I_D, U)^T (I_D, U)(\tilde{x}_t; e_{k_t}) \\ &= -A\cdot Z + c \end{aligned} \quad (17)$$

where $\tilde{x}_t$ denotes a column normalized feature vector, with $\tilde{x}_t := (x_{t1}/\sigma_{k_t 1};\ x_{t2}/\sigma_{k_t 2};\ \ldots;\ x_{tD}/\sigma_{k_t D})$ and

$$A = \frac{1}{2}\sum_{t=1}^{T}(\tilde{x}_t; e_{k_t})(\tilde{x}_t; e_{k_t})^T \quad (18)$$

$$Z = \begin{pmatrix} I_D & U \\ U^T & Y \end{pmatrix} \qquad Y = U^T U \quad (19)$$

Thus, it is straightforward to convert the constraint in eq. (10) into the following form:

$$-d_{ij}(X_i) = \mathcal{F}(X_i|\lambda_j) - \mathcal{F}(X_i|\lambda_{W_i}) = A_{ij}\cdot Z - c_{ij} \le -\rho \quad (20)$$

where $A_{ij} = A_i - A_j$ with A_i and A_j calculated according to eq.(18) based on the Viterbi decoding paths **i** and **j** respectively.

Secondly, we will convert the constraint eq.(11) into the standard SDP form. Similar as above, $R(\Lambda)$ in eq.(8) can be re-written as follows:

$$\begin{aligned} R(\Lambda) &= \sum_{k=1}^{L}(\tilde{\mu}_k - \tilde{\mu}_k^{(0)})^T(\tilde{\mu}_k - \tilde{\mu}_k^{(0)}) \\ &= Q\cdot Z \le r^2 \end{aligned} \quad (21)$$

where $Q = \sum_{k=1}^{n}(\tilde{\mu}_k^{(0)}; e_k)(\tilde{\mu}_k^{(0)}; e_k)^T$, and $\tilde{\mu}_{kd}^{(0)}$ is defined in eq. (16).

Combining eq.(9) with the constraints eq.(20) and eq.(21), we get the following minimization problem:

Problem 2

$$\tilde{\Lambda} = \arg\min_{\Lambda} -\rho \quad (22)$$

subject to

$$A_{ij}\cdot Z + \rho \le c_{ij} \quad \rho \ge 0 \quad (23)$$

$$Q\cdot Z \le r^2 \quad (24)$$

$$Z = \begin{pmatrix} I_D & U \\ U^T & Y \end{pmatrix} \qquad Y = U^T U \quad . \quad (25)$$

for all $X_i \in \mathcal{S}$ and $j \in \Omega$ $j \neq W_i$.

The minimization **Problem 2** is equivalent to the original minimax problem. However, since the constraint $Y = U^T U$ is not convex, it is a non-convex nonlinear optimization problem. As shown in [6], the following statement always holds for matrices:

$$Y - U^T U \succeq 0 \qquad \Leftrightarrow \qquad Z = \begin{pmatrix} I_D & U \\ U^T & Y \end{pmatrix} \succeq 0 \quad (26)$$

Therefore, following [6], if we relax the constraint $Y = U^T U$ to $Y - U^T U \succeq 0$, we are able to make Z a positive semidefinite matrix and in turn convert **Problem 2** into an SDP problem as:

Problem 3

$$\tilde{\Lambda} = \arg\min_{\Lambda} -\rho \quad (27)$$

subject to:

$$A_{ij}\cdot Z + \rho \le c_{ij} \quad (28)$$

$$Q\cdot Z \le r^2 \quad (29)$$

$$Z = \begin{pmatrix} I_D & U \\ U^T & Y \end{pmatrix} \succeq 0 \qquad \rho \ge 0 \quad (30)$$

for all $X_i \in \mathcal{S}$ and $j \in \Omega$ $j \neq W_i$.

Problem 3 is a standard SDP problem, which can be solved efficiently by many SDP algorithms. In problem 3, the optimization is carried out w.r.t. Z (which is constructed from all HMM Gaussian means) and ρ while A_{ij} and c_{ij} and Q are constants calculated from training data, and r is a pre-set parameter. However, due to the relaxation in eq.(30), this SDP problem is just an approximation to the original LME problem. Let us define

$H = Y - U^T U \succeq 0$. After some math manupulations, we can derive that the margin which is maximized in this SDP problem as:

$$\begin{aligned} -d^*_{ij}(X_i) &= A_{i,j} \cdot \begin{pmatrix} I_D & U \\ U^T & U^T U \end{pmatrix} - c_{ij} + A_{i,j} \cdot \begin{pmatrix} 0 & 0 \\ 0 & H \end{pmatrix} \\ &= \frac{1}{2}\sum_{t=1}^{T}(\bar{x}_t - \bar{\mu}_{i_t})^T(\bar{x}_t - \bar{\mu}_{i_t}) - \\ &\quad \frac{1}{2}\sum_{t=1}^{T}(\bar{x}_t - \bar{\mu}_{j_t})^T(\bar{x}_t - \bar{\mu}_{j_t}) - c_{ij} \end{aligned} \quad (31)$$

where $\bar{x}_t := (\tilde{x}_t; 0)$, $\bar{\mu}_{i_t} := (\tilde{\mu}_{i_t}; \sqrt{h_{i_t i_t}})$, and $h_{i_t i_t}$ and $h_{j_t j_t}$ are diagonal elements of H at positions (i_t, i_t) and (j_t, j_t).

Comparing eq. (31) with eqs.(20) and (17), we can see that this SDP problem actually augments each D-dimension speech feature vector x_t to a $(D+1)$-dimensional vector and tries to maximize another margin, $-d^*_{ij}(X_i)$ in eq.(31), in this augmented $(D+1)$-dimension space. At the end, we directly project the optimal solution back to the original D-dimensional space. The SDP algorithms guarantee to find the globally optimal solution in the augmented higher-dimension space, but not the projected one in the original space.

5. EXPERIMENTAL RESULTS

The proposed SDP-based optimization method for LME is evaluated on the TIDIGITS database for continuous speech recognition in the string level[4]. Only adult portion of the corpus is used in our experiments. It contains a total of 225 speakers (111 men and 114 women), 112 of which (55 men and 57 women) are used for training and 113 (56 men, 57 women) for test. The training set has 8623 digit strings and the test set has 8700 strings. Our model set consists of 11 whole-word CDHMMs representing all digits. Each HMM has 12 states and use a simple left-to-right topology without state-skip. Acoustic feature vectors consist of standard 39 dimensions (12 MFCC's and the normalized energy, plus their first and second order time derivatives). Different number of Gaussian mixture components are experimented. We first train models based on maximum likelihood (ML) criterion. Then, MCE training uses the best ML model as the seed model. All HMM model parameters (except transition probabilities) are updated during the MCE training process. At last, we re-estimate the models with the LME method by using both gradient descent and the proposed SDP-based optimization. In LME, we use the best MCE models as the initial models and only HMM mean vectors are re-estimated. In each iteration of LME, a number of competing string-level models are computed for each utterance in training set based on its N-best decoding results ($N = 5$). Then we select support tokens according to eq.(2) and obtain the optimal Viterbi sequence for each support token according to the recognition result. Then, the relaxed SDP optimization , i.e. **Problem 3**, is conducted with respect to Z and ρ. At last, CDHMM means are updated based on the optimization solution through a simple projection. If not convergent, next iteration starts again from recognizing all training data to generate N-Best competing strings. In this work, **Problem 3** is solved by an open software, *DSDP v5.6* [5] running under Matlab.

In Table 1, we give performance comparison in the TIDIGITS test set when using different training criteria to estimate CDHMMs, where LME-Grad represents the LME with the gradient descent method in [3, 4] and LME-SDP represents the LME method with SDP proposed in this paper. It is clearly demonstrated that both LME methods significantly outperform both maximum likelihood (ML) and minimum classification error (MCE) methods. If we compare the SDP method with our previous results based on the gradient descent approach in [4], we can see that the SDP method yields a dramatic improvement in both recognition accuracy and convergence speed. Significant gain has been observed for all model sizes we have examined. This is partly because the SDP method can find the globally optimal solution (not just local optimum) in the augmented higher-dimension space. And the results also show that the approximation caused by relaxation and projection seems reasonably good in the experiments. The best LME model (32-mix) by using SDP method achieves **0.53%** in the string error rate and **0.18%** in WER. To our best knowledge, this is the lowest error rate ever reported in this task.

Table 1. String error rates (in %) on the TIDIGITS test data. (ML: maximum likelihood; MCE: minimum classification error; LME-Grad: LME with gradient descent in [3, 4]; LME-SDP: the proposed SDP-based LME.)

	ML	MCE	LME-Grad	LME-SDP
1-mix	12.61	6.72	3.77	2.75
2-mix	5.26	3.94	1.70	1.24
4-mix	3.48	2.23	1.24	0.89
8-mix	1.94	1.41	0.87	0.68
16-mix	1.72	1.11	0.82	0.63
32-mix	1.34	0.90	0.66	**0.53**

6. SUMMARY

In this paper, we proposed a semidefinite programing (SDP) method for large margin estimation (LME) of CDHMMs in speech recognition. The new optimization method has been demonstrated to be very effective in the TIDIGITS continuous digit string recognition task. Currently, the SDP method is being extended to other large vocabulary continuous speech recognition ASR tasks.

7. REFERENCES

[1] X. Li, H. Jiang and C. Liu, "Large Margin HMMs for Speech Recognition," *Proc. of IEEE ICASSP'2005*, Pennsylvania, Mar. 2005.

[2] C. Liu, H. Jiang and X. Li, "Discriminative Training of CDHMMS for Maximum Relative Separation Margin," *Proc. of IEEE ICASSP'2005*, Pennsylvania, Mar. 2005.

[3] X. Li and H. Jiang, "A Constrained Joint Optimization Method for Large Margin HMM Estimation," *Proc. of 2005 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Nov. 2005.

[4] H. Jiang, X. Li and C. Liu, "Large Margin Hidden Markov Models for Speech Recognition," *to appear in IEEE Trans. on Audio, Speech and Language Processing*, 2006.

[5] S. J. Benson, Y. Ye and X. Zhang, "Solving Large-Scale Sparse Semidefinite Programs for Combinatorial Optimization," SIAM Journal on Optimization, 10(2), 2000, pp. 443-461.

[6] P. Biswas, Y. Ye, "Semidefinite Programming for Ad Hoc Wireless Sensor Network Localization," *Proc. of 2004 Information Processing in Sensor Networks (IPSN)*, Berkeley, 46-54, Apr. 2004.

[7] X. Li, "Large Margin Hidden Markov Models for Speech Recognition," *M.S. thesis*, Department of Computer Science and Engineering, York University, Canada, 2005.